

Method for Accessing Hierarchical Views of a Binary Relational Data-Base

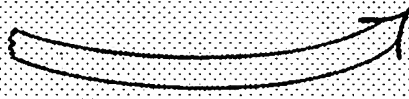
This article describes a method for accessing hierarchical views of data derived from a binary relational database. Access may take place via program operations which are analogous to those supported by a hierarchical database system such as IMS, or via an interpreter which presents hierarchical information on a screen display. In the first case, this would allow application programs written against a hierarchical database to be migrated to run on a binary relational database. In the second case, it would allow end users to obtain a range of presentations which are useful and not easily obtainable from n-ary relational databases. A method for accessing virtual files which are materialized from a binary relational database has been described in [1]. This permits an arbitrary number of virtual flat files to be defined and accessed by means of conventional record operations (i.e., GET, PUT, INSERT, DELETE). In each case, the format of the virtual file is defined in a data dictionary. In this method, an extension to this data dictionary is used to define a "parent-child" relationship between records from two virtual files. Where the key field of one record also appears as a non-key field of the other record, then the second of these may be defined as a "child" of the first. Once this definition has been stored in the data dictionary, it may be used to perform hierarchical retrievals as follows: 1. Retrieve a record from the "parent" virtual file, and extract the value of its key field. 2. Select records from the "child" virtual file such that the relevant non-key field has value equal to that obtained in step 1. 3. Retrieve the first such record from this file. 4. Retrieve the next such record. Repeat this step until no more records are found. 5. Retrieve the next record from the "parent" virtual file, and repeat from step 2. Example of hierarchical retrieval. This procedure implements the "get next" processing which is commonly used with hierarchical databases, and which is useful in displaying or printing reports. It is achieved purely by using already defined operations on flat virtual files. The advantage of this method is that an arbitrary number of hierarchical views may be materialized from a single database. There is therefore great flexibility to define whatever views are needed by application programs or end user presentations, without pre-planning. Application programs also have a high degree of data independence since the physical and logical format of the database may be changed providing that the hierarchical view can still be defined over it. Reference: D. J. Pullin and G. C. H. Sharman, "Virtual File Access to a Binary Relational Database," IBM Technical Disclosure Bulletin 25, 5911-5913 (April 1983).

EMPLOYEE VIRTUAL FILE

EMP#	NAME	DEPT#

DEPARTMENT VIRTUAL FILE

DEPT#	TITLE	LOCN



"child of"

EMP# NAME DEPT

1020	BROWN	23
------	-------	----

5384	JONES	23
------	-------	----

8527	SMITH	23
------	-------	----

2734	GREEN	25
------	-------	----

DEPT# TITLE LOCN

23	SALES	LONDON
----	-------	--------

25	ACCOUNTS	LEEDS
----	----------	-------

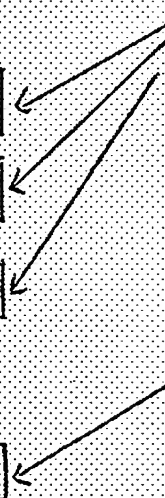


FIG. 2: Example of hierarchical retrieval.